



Higher-order compatible finite element schemes for the nonlinear rotating shallow water equations on the sphere

J. Shipton^{*}, T.H. Gibson, C.J. Cotter

ARTICLE INFO

Article history:

Received 5 July 2017

Received in revised form 16 August 2018

Accepted 17 August 2018

Available online 17 September 2018

Keywords:

Shallow water equations

Numerical weather prediction

Compatible finite elements

Potential vorticity

ABSTRACT

We describe a compatible finite element discretisation for the shallow water equations on the rotating sphere, concentrating on integrating consistent upwind stabilisation into the framework. Although the prognostic variables are velocity and layer depth, the discretisation has a diagnostic potential vorticity that satisfies a stable upwinded advection equation through a Taylor–Galerkin scheme; this provides a mechanism for dissipating enstrophy at the gridscale whilst retaining optimal order consistency. We also use upwind discontinuous Galerkin schemes for the transport of layer depth. These transport schemes are incorporated into a semi-implicit formulation that is facilitated by a hybridisation method for solving the resulting mixed Helmholtz equation. We demonstrate that our discretisation achieves the expected second order convergence and provide results from some standard rotating sphere test problems.

© 2018 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The development of new numerical discretisations based on finite element methods is being driven by the need for more flexibility in mesh geometry. The scalability bottleneck arising from the latitude–longitude grid means that weather and climate model developers are searching for numerical discretisations that are stable and accurate on pseudo-uniform grids without sacrificing properties of conservation, balance and wave propagation that are important for accurate atmosphere modelling on the scales relevant to weather and climate [31]. There is also ongoing interest in adaptively refined meshes as a way of seamlessly coupling global scale and local scale atmosphere simulations, as well as dynamic adaptivity or even moving meshes; using these meshes requires numerical methods that can remain stable and accurate on multiscale meshes. Further, there is an interest in using higher-order spaces to try to offset the inhomogeneity in the error due to using grids that break rotational symmetry.

Compatible finite element methods are a form of mixed finite element methods (meaning that different finite element spaces are used for different fields) that allow the exact representation of the standard vector calculus identities $\text{div-curl}=0$ and $\text{curl-grad}=0$. This necessitates the use of $H(\text{div})$ finite element spaces for velocity, such as Raviart–Thomas and Brezzi–Douglas–Marini, and discontinuous finite element spaces for pressure (stable pairing of velocity and pressure space relies on the existence of bounded commuting projections from continuous to discrete spaces, as detailed in Boffi et al. [9], for example). The main reason for choosing compatible finite element spaces is that they have a discrete Helmholtz decomposition of the velocity space; this means that there is a clean separation between divergence-free and rotational velocity fields. Cotter and Shipton [11] used this decomposition to demonstrate that compatible finite element discretisations for the linear shallow water equations on arbitrary grids satisfy the basic conservation, balance and wave propagation properties listed

^{*} Corresponding author.

E-mail address: jshipton@ic.ac.uk (J. Shipton).

in Staniforth and Thuburn [31]. In particular, it was shown that the discretisation has a geostrophic balancing pressure for every velocity field in the divergence-free subspace of the $H(\text{div})$ finite element space. A survey of the stability and approximation properties of compatible finite element spaces is provided in Natale et al. [25], including a proof of the absence of spurious inertial oscillations.

The challenge of building atmosphere models using compatible finite elements is that there is no freedom to select finite element spaces in order to ensure good representation of the nonlinear equations (such as conservation, or accurate advection, for example), because the choice has already been made to satisfy linear requirements. In the case of the rotating shallow water equations, the use of discontinuous finite element spaces for the layer depth field encourages us to use upwind discontinuous Galerkin methods to solve the continuity equation describing layer depth transport.

The nonlinearity in the momentum/velocity equation is more challenging. In McRae and Cotter [24], the energy-entropy conserving formulation of Arakawa and Lamb [2] was extended to compatible finite element methods. This extension is closely related to C-grid methods for the shallow water equations on more general meshes in Ringle et al. [28], Thuburn and Cotter [32]. Following these approaches, the compatible finite element formulation, which has velocity and height as prognostic variables, has a diagnostic potential vorticity that satisfies a conservation equation that is implied by the prognostic dynamics for velocity and height. A finite element exterior calculus structure in this formulation was exposed in Cotter and Thuburn [12], which also provided an alternative formulation based around low-order finite element methods on dual grids. In Thuburn and Cotter [34], the close relationship of the dual grid formulation to finite volume methods was exploited to obtain a stable discretisation of the nonlinear shallow water equations on the sphere where the finite element formulation of the wave dynamics was coupled with high-order finite volume methods for the layer depth and prognostic potential vorticity fields. The essential idea is to select a particular stable accurate finite volume scheme for the diagnostic potential vorticity, and to then find the update for the prognostic velocity which implies it. In this paper we address the issue of extending this idea to higher-order finite element spaces, for which there is no analogue of the dual grid spaces. This means that we must return to the formulation of McRae and Cotter [24], where the potential vorticity is stored in a continuous finite element space. We then seek stable, accurate higher-order discretisations of the potential vorticity equation using continuous finite element methods that make it possible to find the corresponding update for prognostic velocity. It turns out that this is indeed possible for advection methods from the SUPG/Taylor–Galerkin family of methods.

Finally, we show how these discretisations can be embedded within a semi-implicit time-integration scheme. We again follow the formulation in Thuburn and Cotter [34], in which advection terms are obtained from explicit time integration methods applied using the (iterative) velocity at time level $n + 1/2$. The linear system solved during each nonlinear iteration for the corrections to the field values also requires attention. The standard approach of eliminating velocity to solve a Helmholtz problem for the correction to the layer depth is problematic because the inverse velocity mass matrix is dense. We instead use a hybridised formulation where one solves for the Lagrange multipliers that enforce normal continuity of the velocity field [9, for example].

In section 2 we describe the shallow water model, including the spatial and temporal discretisation; we present finite element spaces that satisfy the properties outlined above and provide details of how to construct such spaces on the sphere and describe advection schemes for both discontinuous and continuous fields as required. In section 3 we present the results of applying our scheme to some of the standard set of test cases for simulation of the rotating shallow water equations on the sphere as described in Williamson et al. [36] and Galewsky et al. [15]. Section 4 provides a summary and brief outlook.

2. The shallow water model

2.1. Shallow water equations

We begin with the vector invariant form of the nonlinear shallow water equations on a two dimensional surface Ω embedded in three dimensions,

$$\mathbf{u}_t + (\zeta + f)\mathbf{u}^\perp + \nabla \left(g(D + b) + \frac{1}{2}|\mathbf{u}|^2 \right) = 0, \quad (1)$$

$$D_t + \nabla \cdot (\mathbf{u}D) = 0, \quad (2)$$

where \mathbf{u} is the horizontal velocity, D is the layer depth, b is the height of the lower boundary, g is the gravitational acceleration, f is the Coriolis parameter and $\zeta = \nabla^\perp \cdot \mathbf{u} := (\mathbf{k} \times \nabla) \cdot \mathbf{u}$ is the vorticity, $\mathbf{u}^\perp = \mathbf{k} \times \mathbf{u}$, \mathbf{k} is the normal to the surface Ω , and where the ∇ and $\nabla \cdot$ operators are defined intrinsically on the surface. These equations have the important property that the shallow water potential vorticity (PV)

$$q = \frac{\zeta + f}{D} \quad (3)$$

satisfies a local conservation law,

$$\frac{\partial}{\partial t}(Dq) + \nabla \cdot (\mathbf{u}qD) = 0. \quad (4)$$

This can be seen by applying $\nabla^\perp \cdot$ to equation (1). Equation (2) then implies that q is constant along characteristics moving with the flow velocity \mathbf{u} , i.e.,

$$\frac{\partial q}{\partial t} + (\mathbf{u} \cdot \nabla) q = 0. \quad (5)$$

Numerical discretisations that preserve some aspects of these properties have been demonstrated to be very successful at obtaining long time integrations of the rotating shallow water equations on sphere in the quasi-geostrophic flow regime. This is partially because they provide a way to avoid discretising the vector advection term $(\mathbf{u} \cdot \nabla)\mathbf{u}$ in the velocity equation directly; instead one can choose a suitable stable, accurate and conservative scalar advection scheme for the potential vorticity (treated as a diagnostic variable with \mathbf{u} and D being prognostic variables) and use it to diagnose a form of the $(\zeta + f)\mathbf{u}^\perp$ term in equation (1) that leads to stable advection of \mathbf{u} . These ideas were introduced in the compatible finite element context in McRae and Cotter [24] in order to obtain energy-entropy conserving discretisations; here we concentrate on stable, accurate and conservative advection of q , improving on the low-order APVM stabilisation suggested there. We also replace the centred discretisation of equation (2) with stable and accurate Discontinuous Galerkin (DG) advection schemes for D , and show how these can be incorporated into the PV conserving formulation.

2.2. Spatial discretisation

2.2.1. Finite element spaces

In this section we shall summarise the properties we require from our finite element spaces and the operators between them. We start with the space $H(\text{div})$ of square integrable velocity fields, whose divergence is also square integrable. The condition that the discrete velocity belongs to the finite element subspace $\mathbb{V}_1 \subset H(\text{div})$ means that the velocity must have continuous normal components across element edges. Having chosen \mathbb{V}_1 , we select a finite element space $\mathbb{V}_2 \subset L^2$ such that

$$\{\nabla \cdot \mathbf{w} : \mathbf{w} \in \mathbb{V}_1\} \subset \mathbb{V}_2. \quad (6)$$

This necessarily requires that \mathbb{V}_2 is a discontinuous space. We also define a space $\mathbb{V}_0 \subset H^1$ consisting of continuous fields γ such that $\mathbf{k} \times \nabla \psi \in \mathbb{V}_1$, where the curl $\mathbf{k} \times \nabla$, henceforth written as ∇^\perp , maps from \mathbb{V}_0 onto the kernel of $\nabla \cdot$ in \mathbb{V}_1 .

The proof that the mixed finite element discretisation of the linear shallow water equations has steady geostrophic modes relies on the existence of a discrete Helmholtz decomposition for the velocity field [11]. As described in Arnold et al. [3], this decomposition exists if the following diagram commutes with bounded projections π_0, π_1, π_2 ,

$$\begin{array}{ccccc} H^1 & \xrightarrow{\nabla^\perp} & H(\text{div}) & \xrightarrow{\nabla \cdot} & L^2 \\ \downarrow \pi_0 & & \downarrow \pi_1 & & \downarrow \pi_2 \\ \mathbb{V}_0 & \xrightarrow{\nabla^\perp} & \mathbb{V}_1 & \xrightarrow{\nabla \cdot} & \mathbb{V}_2 \end{array} \quad (7)$$

that is, the result of applying an operator to the continuous field and projecting into the discrete space is the same as the result of first projecting the field into the discrete space and then applying the operator.

Cotter and Shipton [11] reviewed several sets of finite element spaces that satisfy these requirements, together with further requirements on the degree-of-freedom (DOF) ratios between \mathbb{V}_1 and \mathbb{V}_2 that are necessary to exclude the possibility of spurious mode branches in the dispersion relation for the linear shallow water equations. In this paper we shall present results from choosing \mathbb{V}_1 to be the first order Brezzi–Douglas–Marini (BDM₂) function space on triangles which requires that \mathbb{V}_0 is P_3 , the space of piecewise continuous cubic functions. \mathbb{V}_2 is the space P_1^{DG} of piecewise linear functions that can be discontinuous at the element boundaries.

2.2.2. Constructing the finite element spaces on the sphere

In order to implement the finite element method, we need to expand our variables in terms of suitable basis functions and compute integrals of combinations of these basis functions over elements. This is done by defining a reference element on which these integrals can be calculated and mappings from each reference element to the physical element, which we describe in this section. A more detailed and general exposition, plus description of how these mappings are implemented in the FEniCS project, is provided in [29]. In the rest of this section, hatted quantities refer to those defined on the reference element.

We start by defining finite element spaces $\mathbb{V}_i(\hat{e})$, $i = 0, 1, 2$, on the reference element \hat{e} . These are constructed from polynomials in the usual manner for the chosen spaces as described in [9], for example. Then, for each element e , we construct a polynomial mapping $\mathbf{g}_e : \hat{e} \mapsto \mathbb{R}^3$, such that

$$\hat{\Omega} = \bigcup_{i=1}^{N_e} \mathbf{g}_{e_i}(\hat{e}), \quad \mathbf{g}_{e_1}(\hat{e}) \cap \mathbf{g}_{e_2}(\hat{e}) = \emptyset, \quad \text{if } e_1 \neq e_2, \quad (8)$$

where $\hat{\Omega}$ is the computational domain, which is a piecewise polynomial approximation to the sphere domain Ω . Then, we use \mathbf{g}_e to relate functions in our finite element spaces on $\hat{\Omega}$, restricted to each element e in $\hat{\Omega}$, to functions in $\mathbb{V}_i(\hat{e})$.

For $\mathbb{V}_0, \mathbb{V}_2$, these functions are simply related by function composition, i.e.,

$$\psi \in \mathbb{V}_0 \implies \psi|_e \circ \mathbf{g}_e \in \mathbb{V}_0(\hat{e}), \quad \phi \in \mathbb{V}_2 \implies \phi|_e \circ \mathbf{g}_e \in \mathbb{V}_2(\hat{e}), \quad (9)$$

For \mathbb{V}_1 , we use the Piola transformation [9]

$$\mathbf{u}|_e(\mathbf{g}_e(\hat{\mathbf{x}})) = \frac{J_e \hat{\mathbf{u}}}{\det J_e}, \quad (10)$$

where $\mathbf{u}|_e$ is the restriction of \mathbf{u} to element e , and

$$J_e = \frac{\partial \mathbf{g}_e}{\partial \hat{\mathbf{x}}} \quad (11)$$

is the Jacobian of \mathbf{g} in element e , to ensure that \mathbf{u} is tangent to Ω . The Piola transformation has the crucial property that

$$\nabla \cdot \mathbf{u}(\mathbf{g}_e(\hat{\mathbf{x}})) = \frac{\hat{\nabla} \cdot \hat{\mathbf{u}}(\hat{\mathbf{x}})}{\det J_e(\hat{\mathbf{x}})}. \quad (12)$$

When the discrete space is made up of flat elements, as in Cotter and Shipton [11] and McRae and Cotter [24], $\det J_e$ is constant. This guarantees that if $\hat{\nabla} \cdot \hat{\mathbf{u}} \in \hat{\mathbb{V}}_1$ then $\nabla \cdot \mathbf{u} \in \mathbb{V}_1$. However, for meshes made up of general quadrilaterals (i.e. cubed sphere meshes such as those in Putman and Lin [26]) or high-order, curved triangles, the mapping \mathbf{g} is no longer affine and the determinant of its Jacobian is no longer constant. This means that in general $\nabla \cdot \mathbf{u} \notin \mathbb{V}_1$. This clearly violates the commutative diagram property (7). There are 3 ways to remedy the situation.

1. Modify the mapping for \mathbb{V}_2 to become

$$\phi \in \mathbb{V}_2 \implies \det J_e \phi|_e \circ \mathbf{g}_e \in \mathbb{V}_2(\hat{e}). \quad (13)$$

This option is the choice that is most consistent with the finite element exterior calculus methodology.

2. Modify the mapping for \mathbb{V}_1 so that the factor of $\det J_e$ in $\nabla \cdot \mathbf{u}|_e \circ \mathbf{g}_e$ is replaced by the element average of $\det J_e$. This approach was described in Boffi and Gastaldi [8] for the case of lowest order Raviart–Thomas elements; the extension to other H(div) elements is straightforward but the construction is quite complicated.
3. Replace the divergence operator $\nabla \cdot$ in the commutative diagram by $\tilde{\nabla} \cdot$ to be the L_2 projection π_2 of $\nabla \cdot$ into \mathbb{V}_2 , defined by

$$\int_{\Omega} \phi \tilde{\nabla} \cdot \mathbf{u} \, dV = \int_{\Omega} \phi \nabla \cdot \mathbf{u} \, dV, \quad \forall \phi \in \mathbb{V}_2. \quad (14)$$

By defining a new divergence operator in this way we immediately recover the required commutation property, since π_2 appears in the commutative diagram (7) and is a projection. This is a generalisation of the “rehabilitation” technique described for lowest order Raviart–Thomas elements applied to mixed elliptic problems in [7]. In fact, as Bochev and Ridzal noticed, introduction of the $\tilde{\nabla} \cdot$ operator does not require any changes to a code implementation for mixed elliptic operators since the $\tilde{\nabla} \cdot$ operator only appears in an inner product with a test function from \mathbb{V}_2 , and hence can be safely replaced by $\nabla \cdot$ there. We shall see that this continues to be the case in our nonlinear shallow water formulation.

In general, Option 1 is the most mathematically elegant choice. However, since our software implementation is based upon the Firedrake finite element library [27,23] which is already used to develop DG schemes that use the transformations in (9), Option 3 was much less pervasive through the code base. We will investigate the differences between Options 1 and 3 in future work.

Arnold et al. [4] showed that there is a potential problem with loss of consistency with compatible finite elements on quadrilateral and cubic curvilinear cells; it is likely that this problem also can be exhibited on triangular cells. In particular, for general constructions there may be loss of consistency for the \mathbb{V}_1 spaces used here (the consistency problem for \mathbb{V}_2 is avoided through rehabilitation). However, Holst and Stern [20] demonstrated that consistent approximation can still be obtained if the computational domain can be obtained via a transformation from a mesh of affine elements onto the higher-order nodal interpolation of a C^∞ manifold (such as the sphere). This aspect is discussed further in the context of geophysical fluid dynamics in [25].

2.2.3. Mixed finite element formulation

The mixed finite element discretisation of equations (1)–(2) is formed by restricting $\mathbf{u} \in \mathbb{V}_1$, $D \in \mathbb{V}_2$, multiplying the equations by appropriate test functions, $\mathbf{w} \in \mathbb{V}_1$ and $\phi \in \mathbb{V}_2$, and integrating over the domain, giving

$$\int_{\Omega} \mathbf{w} \cdot \mathbf{u}_t \, dV + \int_{\Omega} \mathbf{w} \cdot \mathbf{Q}^\perp \, dV - \int_{\Omega} \nabla \cdot \mathbf{w} \left(g(D+b) + \frac{1}{2} |\mathbf{u}|^2 \right) \, dV = 0, \quad \forall \mathbf{w} \in \mathbb{V}_1, \quad (15)$$

$$\int_{\Omega} \phi \left(D_t + \tilde{\nabla} \cdot \mathbf{F} \right) \, dV = 0, \quad \forall \phi \in \mathbb{V}_2, \quad (16)$$

where we have introduced the mass flux $\mathbf{F} \in \mathbb{V}_1$ (an approximation to $\mathbf{u}D$), and the vorticity flux \mathbf{Q} (an approximation to $\mathbf{u}dq$). \mathbf{F} and \mathbf{Q} are defined by appropriate choice of advection schemes for D and the diagnostic potential vorticity q , which we shall define later.

We have integrated the gradient term in equation (15) by parts to avoid taking the gradient of the layer depth which is undefined since $D \in \mathbb{V}_2$ is discontinuous. A similar problem is posed for the definition of the potential vorticity q since $\zeta = \nabla^\perp \cdot \mathbf{u}$, which appears in the definition of q , is similarly undefined. In order to fix this we integrate the ∇^\perp term by parts (with no surface term since we are solving our equations on the sphere), defining our discrete potential vorticity $q \in \mathbb{V}_0$ as

$$\int_{\Omega} \gamma q D \, dV = \int_{\Omega} -\nabla^\perp \gamma \cdot \mathbf{u} \, dV + \int_{\Omega} \gamma f \, dV, \quad \forall \gamma \in \mathbb{V}_0. \quad (17)$$

Provided D remains positive (this can be enforced using a slope limiter, although in the test cases used here the mean depth is sufficiently high that a slope limiter is not required), then this equation can be solved for $q \in \mathbb{V}_0$ from known D and \mathbf{u} . This provides a one-to-one mapping between q and the weak curl of \mathbf{u} , which leads to a one-to-one mapping between q and the divergence-free part of \mathbf{u} via the discrete Helmholtz decomposition (without needing to solve a global Poisson problem). Hence, control of q in the L^2 norm from a chosen stable advection scheme provides strong control over the divergence-free component of \mathbf{u} without compromising the divergent part. This is one reason for the success of this type of potential vorticity conserving discretisation for the rotating shallow water equations.

We differentiate this equation in time, and substitute for \mathbf{u}_t using equation (15) with $\mathbf{w} = -\nabla^\perp \gamma$. Since $\nabla \cdot \nabla^\perp \equiv 0$ and we assume $f_t = 0$, this gives

$$\int_{\Omega} \gamma (qD)_t \, dV - \int_{\Omega} \nabla \gamma \cdot \mathbf{Q} \, dV = 0, \quad \forall \gamma \in \mathbb{V}_0, \quad (18)$$

which is the Galerkin projection of the PV conservation law into \mathbb{V}_0 . If we select $\gamma = 1$, then we obtain conservation of total PV,

$$\frac{d}{dt} \int_{\Omega} qD \, dV = 0. \quad (19)$$

On the other hand, if we are on the sphere, then this quantity is zero as can be computed directly from (17); this topological result stems from the fact that in this case $\mathbf{w} = -\nabla^\perp \gamma = 0$.

If we choose our advection scheme so that $\mathbf{Q} = q\mathbf{F}$ and if q is a constant, then we may integrate (18) by parts without introducing error (since $q \in H^1$ and $\mathbf{F} \in H(\text{div})$), and we obtain

$$\int_{\Omega} \gamma (qD)_t \, dV = - \int_{\Omega} \gamma q \nabla \cdot \mathbf{F} \, dV, \quad (20)$$

and hence

$$\int_{\Omega} \gamma D q_t \, dV = - \int_{\Omega} \gamma q (D_t + \nabla \cdot \mathbf{F}) \, dV. \quad (21)$$

For flat elements, or if we had chosen Option 1, the right hand side is zero since D and $\nabla \cdot \mathbf{F}$ are in the same finite element space and therefore equation (16) holds pointwise, i.e.,

$$D_t + \nabla \cdot \mathbf{F} = 0. \quad (22)$$

Hence, we conclude that if q is constant, then $q_t = 0$, and so q remains constant. As well as being a statement of first-order consistency for the advection scheme for q , this is also an important property of equation (5).

The formulation requires some adaptation for the finite element spaces used in this paper to obtain this result, since $\nabla \cdot \mathbf{F}$ is not guaranteed to be in the same space as D , so equation (22) does not hold pointwise. To recover it, some further “rehabilitation” is required; we amend equation (17) by defining $\tilde{D} \in \mathbb{V}_2$ such that

$$\frac{\tilde{D}_t}{\tau} + \nabla \cdot \mathbf{F} = 0, \quad (23)$$

where

$$\tau|_e = \det J_e \circ \mathbf{g}_e. \quad (24)$$

Comparing the weak form of this equation with (16) we see that

$$\int_{\Omega} \phi \frac{\tilde{D}_t}{\tau} dV = \int_{\Omega} \phi D_t dV, \quad \forall \phi \in \mathbb{V}_2, \quad (25)$$

which is consistent with the definition

$$\int_{\Omega} \phi \frac{\tilde{D}}{\tau} dV = \int_{\Omega} \phi D dV \quad \forall \phi \in \mathbb{V}_2, \quad (26)$$

hence we can solve for \tilde{D} (since τ is a positive quantity and just alters the metric). \tilde{D} is discontinuous and hence this equation can be solved separately in each element.

Using this in equation (17) gives

$$\int_{\Omega} \gamma q \frac{\tilde{D}}{\tau} dV = - \int_{\Omega} \nabla^\perp \gamma \cdot \mathbf{u} dV + \int_{\Omega} \gamma f dV \quad \forall \gamma \in \mathbb{V}_0. \quad (27)$$

Differentiating, rearranging and assuming q is constant as before we obtain

$$\int_{\Omega} \gamma \frac{\tilde{D}}{\tau} q_t dV = - \int_{\Omega} \gamma q \left(\frac{\tilde{D}_t}{\tau} + \nabla \cdot \mathbf{F} \right) dV = 0, \quad (28)$$

as required.

In fact, the finite element formulation allows us to go beyond the first order consistency result. For example, if we choose $\mathbf{Q} = \mathbf{F}q$, we have enough continuity to integrate by parts, and we obtain

$$\int_{\Omega} \gamma ((qD)_t + \nabla \cdot (\mathbf{F}q)) dV = 0, \quad \forall \gamma \in \mathbb{V}_0. \quad (29)$$

The left-hand side vanishes if q is an exact solution of the equation

$$(qD)_t + \nabla \cdot (\mathbf{F}q) = 0, \quad (30)$$

where D and \mathbf{F} are the discrete mass and mass flux, indicating that the discretisation is consistent at the order of approximation of the finite element space \mathbb{V}_0 . Unfortunately, this is not a good choice for \mathbf{Q} since the implied discrete advection operator is not stable. An alternative is to choose

$$\mathbf{Q} = \mathbf{F}q - \alpha \mathbf{F} \frac{\mathbf{F}}{|\mathbf{F}|} \cdot \nabla q, \quad (31)$$

where α is a stabilisation parameter. Substitution, rearrangement and integration by parts on the $\mathbf{F}q$ term leads to

$$\int_{\Omega} \left(\gamma + \alpha \frac{\mathbf{F}}{|\mathbf{F}|} \cdot \nabla \gamma \right) ((qD)_t + \nabla \cdot (\mathbf{F}q)) dV = 0, \quad \forall \gamma \in \mathbb{V}_0, \quad (32)$$

which is a streamline-upwind Petrov–Galerkin (SUPG) spatial discretisation of the PV conservation equation. This discretisation is stable, and also consistent, *i.e.* the equation vanishes when the exact solution to (30) is substituted.

In this paper we will use a Taylor–Galerkin discretisation for the potential vorticity equation; this discretisation achieves the same aims as the SUPG discretisation described above, but arises more naturally in the discrete time setting and hence we shall postpone our discussion of it until we have described the time-discrete formulation of the full shallow water system in the next section.

We remark that all of the results above are assuming that integrations are evaluated exactly. Due to the factors of $1/\det J$ arising from the Piola transformation, not all integrands are polynomial and hence exact quadrature is difficult. However, as noted in [12], the structure of terms of the following forms,

$$\int_{\Omega} \nabla g \cdot \mathbf{v} \, dx, \quad \int_{\Omega} g \mathbf{w} \cdot \mathbf{v}^{\perp} \, dx, \quad \int_{\Omega} g \nabla \cdot \mathbf{w} \, dx, \quad (33)$$

where g is an arbitrary scalar function (such as a product of scalar functions) and \mathbf{w} and \mathbf{v} are Piola-mapped functions, means that the factors of $\det J$ cancel after transforming to the reference element, and the integral after change of variables has a polynomial integrand after all. Later on this will also apply to integrals of the form

$$\int_f \mathbf{v} \cdot \mathbf{n} g \, dS, \quad (34)$$

where f is an element facet. This means that all of the conservation properties in this paper also hold for inexact quadrature provided that it is sufficiently high order to integrate these polynomial integrands exactly, after appropriately redefining inner products using a quadrature rule instead of an exact integral.

2.3. Time discretisation

We shall build a semi-implicit time discrete formulation. First we write

$$\mathbf{u}^* = \theta \mathbf{u}^{n+1} + (1 - \theta) \mathbf{u}^n, \quad D^* = \theta D^{n+1} + (1 - \theta) D^n \quad (35)$$

and

$$\Delta \mathbf{u} = \mathbf{u}^{n+1} - \mathbf{u}^n, \quad \Delta D = D^{n+1} - D^n. \quad (36)$$

We can now write equations (15) and (16) as

$$\int_{\Omega} \mathbf{w} \cdot \Delta \mathbf{u} \, dV + \Delta t \int_{\Omega} \mathbf{w} \cdot \mathbf{Q}^{\perp} \, dV - \Delta t \int_{\Omega} \nabla \cdot \mathbf{w} (g(D^* + b) + K(\mathbf{u}^*)) \, dV = 0, \quad \forall \mathbf{w} \in \mathbb{V}_1, \quad (37)$$

$$\int_{\Omega} \phi \Delta D \, dV + \Delta t \int_{\Omega} \phi \nabla \cdot \mathbf{F} \, dV = 0, \quad \forall \phi \in \mathbb{V}_2, \quad (38)$$

where

$$K(\mathbf{u}) = \frac{1}{2} |\mathbf{u}|^2, \quad (39)$$

and where the time-averaged mass and vorticity fluxes \mathbf{F} and \mathbf{Q} are yet to be defined. The idea is that we choose \mathbf{F} to be a time-independent function such that D^{n+1} is obtained from D^n via a high-order stable time discretisation over one timestep for the equation

$$D_t + \nabla \cdot (\mathbf{u}^* D) = 0, \quad (40)$$

i.e. with the advecting velocity frozen to the value of \mathbf{u}^* . Similarly, \mathbf{Q} is chosen so that q^{n+1} is related to q^n via a high-order stable time discretisation over one timestep for the equation

$$(qD)_t + \nabla \cdot (\mathbf{F}q) = 0, \quad (41)$$

i.e. with the advecting mass flux frozen to the time-averaged flux \mathbf{F} . This means that for $\theta = 1/2$ we obtain a scheme that is overall second-order in time, but that uses higher-order advection schemes for D and q . The rationale is that for near-linear waves, we would like the propagation to be as conservative as possible, so the semi-linear formulation should be based around a time-centred scheme. However, we would also like to obtain good quality solutions over long integrations when close to geostrophic balance, in which case the important quantity is q , and it is important that we transport D consistently with q to stay close to the balanced state. In that regime, it is thought to be important to use a high odd-order time integration scheme, since for odd-order schemes the error is dominated by diffusion rather than dispersion (the latter leads to oscillations near to near-discontinuous data). It is also thought that the use of a time-averaged velocity to transport q and D helps to preserve geostrophic balance. This was also the rationale behind choosing the 3rd order Forward-in-Time advection schemes used in Thuburn and Cotter [34].

An implicit formulation such as the one above requires Newton or Picard iterations to iterate to convergence. In practice, we perform a small fixed number (4) of Picard iterations per timestep, since our aim is to obtain a stable timestepping method with accurate q and D transport, rather than iterating to convergence to obtain an exact implementation of the fully implicit scheme described above. In the Picard iteration we replace the Jacobian obtained from linearisation around the current state with the Jacobian linearised around the system at a state of rest. This means that it is possible to reduce the system as described in the next section, and that the same solver context can be reused during each Picard iteration and timestep.

Hence, each Picard cycle consists of the following steps.

1. Initialise $\Delta \mathbf{u} = 0$, $\Delta D = 0$.
2. Use the current values of $\Delta \mathbf{u}$ and ΔD to compute corresponding values for \mathbf{u}^* , D^* .
3. Use the chosen mass advection scheme with velocity \mathbf{u}^* to update from D^n to D^{n+1} , and find \mathbf{F} such that

$$\int_{\Omega} \phi (D^{n+1} - D^n + \Delta t \nabla \cdot \mathbf{F}) dV = 0, \quad \forall \phi \in \mathbb{V}_2. \quad (42)$$

We will explain how to construct \mathbf{F} in section 2.3.2. The mass residual $R_D : \mathbb{V}_2 \rightarrow \mathbb{R}$ is defined as

$$R_D[\phi] = \int_{\Omega} \phi (\Delta D + \Delta t \nabla \cdot \mathbf{F}) dV. \quad (43)$$

4. Diagnose the PV q^n at time t^n .
5. Use the chosen PV advection scheme with mass flux \mathbf{F} to update from q^n to q^{n+1} and compute the corresponding PV flux \mathbf{Q} such that

$$\int_{\Omega} \gamma \left(q^{n+1} \frac{\tilde{D}^{n+1}}{\tau} - q^n \frac{\tilde{D}^n}{\tau} \right) dV - \Delta t \int_{\Omega} \nabla \gamma \cdot \mathbf{Q} dV = 0, \quad \forall \gamma \in \mathbb{V}_0. \quad (44)$$

6. The velocity residual $R_{\mathbf{u}} : \mathbb{V}_1 \rightarrow \mathbb{R}$ is defined as

$$R_{\mathbf{u}}[\mathbf{w}] = \int_{\Omega} \mathbf{w} \cdot \Delta \mathbf{u} dV + \Delta t \int_{\Omega} \mathbf{w} \cdot \mathbf{Q}^{\perp} dV + \Delta t \int_{\Omega} \nabla \cdot \mathbf{w} (g(D^* + b) + K(\mathbf{u}^*)) dV. \quad (45)$$

7. The increments

$$\Delta \mathbf{u} \mapsto \Delta \mathbf{u} + \delta \mathbf{u}, \quad \Delta D \mapsto \Delta D + \delta D \quad (46)$$

are then obtained by solving the coupled system

$$\int_{\Omega} \mathbf{w} \cdot \delta \mathbf{u} dV + \theta \Delta t \int_{\Omega} f \mathbf{w} \cdot \delta \mathbf{u}^{\perp} dV - \theta \Delta t \int_{\Omega} \nabla \cdot \mathbf{w} g \delta D dV = -R_{\mathbf{u}}[\mathbf{w}], \quad \forall \mathbf{w} \in \mathbb{V}_1, \quad (47)$$

$$\int_{\Omega} \phi (\delta D + \theta \Delta t H_0 \nabla \cdot \delta \mathbf{u}) dV = -R_D[\phi], \quad \forall \phi \in \mathbb{V}_2, \quad (48)$$

where H_0 is the mean layer depth at rest.

8. If we have not completed 4 iterations, apply these updates and return to 2.

In the following sections we describe the construction of \mathbf{F} , \mathbf{Q} and the solution of the coupled system in detail.

2.3.1. Solving the coupled linear system

In our formulation, we use hybridisation to solve equations (47)–(48). Hybridisation is a technique for efficiently solving mixed finite element problems that has been used since the 1960s; in the 1980s it was also discovered that the hybridised formulation could also be used to obtain more accurate approximations of the solution (see [9] for a general survey).

Obtaining a hybridised formulation requires two steps. First, we introduce a finite element space $\tilde{\mathbb{V}}_1$ by relaxing the normal continuity constraints within \mathbb{V}_1 . In other words, functions in $\tilde{\mathbb{V}}_1$ have the same local polynomial representation as functions in \mathbb{V}_1 , but there are no requirements of continuity between edges. In particular, we note that $\mathbb{V}_1 \subset \tilde{\mathbb{V}}_1$. Second, we introduce a trace space $\text{Tr}(\mathbb{V}_1)$, defined on the element facet set Γ , such that functions $\lambda \in \text{Tr}(\mathbb{V}_1)$ are scalar functions which when restricted to a single element facet f , are from the same polynomial space as $\mathbf{u} \cdot \mathbf{n}$ restricted to that facet. Having relaxed the continuity requirements for $\delta \mathbf{u} \in \tilde{\mathbb{V}}_1$, we enforce them again by adding another equation,

$$\int_{\Gamma} \mu [\![\mathbf{u}]\!] dS = 0, \quad \forall \mu \in \text{Tr}(\mathbb{V}_1), \quad (49)$$

where we use the usual “jump” notation

$$[\![\mathbf{u}]\!] = \mathbf{u}^+ \cdot \mathbf{n}^+ + \mathbf{u}^- \cdot \mathbf{n}^-, \quad (50)$$

having arbitrarily labelled each side of each facet with + and –, so that \mathbf{n}^+ points from the + side to the – side and vice versa. To avoid an over-determined system, we introduce Lagrange multipliers $\lambda \in \text{Tr}(\mathbb{V}_1)$ and rewrite equations (47)–(48) as

$$\int_{\Omega} \mathbf{w} \cdot \delta \mathbf{u} \, dV + \theta \Delta t \int_{\Omega} f \mathbf{w} \cdot \mathbf{u}^{\perp} \, dV - \theta \Delta t \int_{\Omega} \nabla \cdot \mathbf{w} g \delta D \, dV + \int_{\Gamma} \lambda \llbracket \mathbf{w} \rrbracket \, dS = -R_{\mathbf{u}}[\mathbf{w}], \quad \forall \mathbf{w} \in \tilde{\mathbb{V}}_1, \quad (51)$$

$$\int_{\Omega} \phi (\delta D + \theta \Delta t H_0 \nabla \cdot \delta \mathbf{u}) \, dV = -R_D[\phi], \quad \forall \phi \in \mathbb{V}_2, \quad (52)$$

together with equation (49). Note that the residual $R_{\mathbf{u}}$ must now be evaluated with $\mathbf{w} \in \tilde{\mathbb{V}}_1$. All of the inter-element coupling in equations (51)–(52) takes place in the λ term. This means that if λ is known, then it is possible to obtain \mathbf{u} and D independently in each element. To enable elimination of \mathbf{u} and D , we define a lifting operator $L : \text{Tr}(\mathbb{V}_1) \rightarrow \tilde{\mathbb{V}}_1$, which gives the solution \mathbf{u} for a given λ in the case when $R_D[\phi]$ and $R_{\mathbf{u}}[\mathbf{w}]$ are replaced by zero. We also define \mathbf{u}_0 which is the solution obtained when λ is zero, but $R_D[\phi]$ and $R_{\mathbf{u}}[\mathbf{w}]$ are present. Then, the general solution of this equation given particular R_D and $R_{\mathbf{u}}$ is

$$\mathbf{u} = L\lambda + \mathbf{u}_0. \quad (53)$$

Substituting into equation (49) gives

$$\int_{\Gamma} \mu \llbracket L\lambda \rrbracket \, dS = - \int_{\Gamma} \mu \llbracket \mathbf{u}_0 \rrbracket \, dS, \quad \forall \mu \in \text{Tr}(\mathbb{V}_1). \quad (54)$$

This reduced equation can be solved for λ before reconstructing \mathbf{u} and D by solving equations (51)–(52) independently in each element. Since the value of $L\lambda$ in each element only depends on the values of λ on the facets of that element, equation (54) only couples together values of λ on facets that share an element. This means that the matrix-vector form of this equation is very sparse. In fact, the matrix can be assembled by visiting each element separately, performing inversion on element block systems. Further, this equation has the same spectral properties as the Helmholtz operator, and hence can be solved with Krylov methods and standard preconditioners such as SOR, algebraic multigrid; a geometric multigrid for general higher-order hybridised mixed finite element elliptic problems was provided in [17]. Note that the Coriolis term can be included in the linear system in this approach without altering the sparsity of the reduced system. One important aspect is that if the solver for this system is not iterated to convergence, the resulting velocity field will not be exactly div-conforming. We address this in our implementation by simply projecting the velocity back into \mathbb{V}_1 after reconstruction, which appears not to cause any problems with stability. A more sophisticated approach would use the hybridised solver as a preconditioner for the $(\Delta u, \Delta D)$ system on $(\mathbb{V}_1, \mathbb{V}_2)$; we will investigate this in further work.

2.3.2. Advection scheme for layer depth D

We now need to solve the mass continuity equation (16) for the update to D . As $D \in \mathbb{V}_2$ and is therefore discontinuous, we can use standard upwind discontinuous Galerkin methods to obtain an approximation to D_t ,

$$\int_e \phi D_t \, dV = \Delta t \int_e \nabla \phi \cdot \mathbf{u}^* D \, dV - \Delta t \int_{\partial e} \phi D^u \mathbf{u}^* \cdot \mathbf{n} \, dS, \quad \forall \phi \in \mathbb{V}_2(e), \quad (55)$$

for each element e , where $\mathbb{V}_2(e)$ is the space \mathbb{V}_2 restricted to the element e , and D^u is the value of D on the upwind side of the element boundary ∂e . We then use the standard 3-stage Strong Stability Preserving Runge–Kutta scheme [18],

$$D^1 = D^n + \Delta t D_t^n, \quad (56)$$

$$D^2 = \frac{3}{4} D^n + \frac{1}{4} (D^1 + \Delta t D_t^1), \quad (57)$$

$$D^{n+1} = \frac{1}{3} D^n + \frac{2}{3} (D^2 + \Delta t D_t^2). \quad (58)$$

Later we shall discuss a consistency property of the potential vorticity conserving discretisation of the velocity equation; this property requires that we find a time-integrated mass flux $\bar{\mathbf{F}} \in \mathbb{V}_1$ that satisfies

$$\Delta D = -\Delta t \nabla \cdot \bar{\mathbf{F}}. \quad (59)$$

This is done by finding \mathbf{F} given D_t such that

$$D_t + \nabla \cdot \mathbf{F} = 0, \quad (60)$$

and then substituting into equations (56)–(58) to construct $\bar{\mathbf{F}}$. To find a flux \mathbf{F} for each Runge–Kutta stage, we solve

$$\int_f \kappa \mathbf{F} \cdot \mathbf{n} \, ds = \int_f \kappa D^u \mathbf{u}^* \cdot \mathbf{n} \, ds, \quad \forall \kappa \in \text{Tr}_f(\mathbb{V}_1), \quad \forall f \in \partial e \quad (61)$$

$$\int_e \mathbf{w} \cdot \mathbf{F} \, dx = \int_e \mathbf{w} \cdot \mathbf{u}^* D \, dx, \quad \forall \mathbf{w} \in \mathbb{V}_1^*(e), \quad (62)$$

where $\mathbb{V}_1^*(e)$ is the right size to close the system and $\nabla \phi \in \mathbb{V}_1^*(e)$ for all $\phi \in \mathbb{V}_2(e)$. The left-hand sides of equations (61)–(62) are the same as the left-hand sides of the definition of the commuting operator $\pi : H(\text{div}) \rightarrow \mathbb{V}_1$ that features in stability proofs for mixed finite element methods (see [9], for example). This means that the above construction is well-posed. To check that equation (60) is satisfied, we integrate (59) by parts. Then, substituting the above relations (61)–(62) (with $\mathbf{w} = \nabla \phi$), we see that

$$\begin{aligned} \int_e \phi D_t \, dV &= -\Delta t \int_e \phi \nabla \cdot \mathbf{F} \, dV \\ &= \Delta t \int_e \nabla \phi \cdot \mathbf{F} \, dV - \Delta t \int_{\partial e} \phi \mathbf{F} \cdot \mathbf{n} \, dS \\ &= \Delta t \int_e \nabla \phi \cdot \mathbf{u}^* D \, dV - \Delta t \int_{\partial e} \phi D^u \mathbf{u}^* \cdot \mathbf{n} \, dS, \end{aligned}$$

as required.

We note, although we did not use it in this paper, that the use of a slope limiter can also be incorporated into the mass flux computation. If a slope limiter is used after a Runge–Kutta stage, then D is replaced by D' , with $\int_e D - D' \, dx = 0$. If we seek \mathbf{F}' such that

$$D' - D + \Delta t \nabla \cdot \mathbf{F}' = 0, \quad (63)$$

then integration over a single element immediately tells us that this can be satisfied by

$$\int_f \kappa \mathbf{F}' \cdot \mathbf{n} \, dS = 0, \quad \forall \kappa \in \text{Tr}_f(\mathbb{V}_1), \quad \forall f \in \partial e, \quad (64)$$

i.e. $\mathbf{F}' \cdot \mathbf{n} = 0$ on the boundary ∂e . We then solve a local mixed problem for $(\mathbf{F}', p) \in (\mathbb{V}_1(e), \mathbb{V}_2(e))$, given by

$$\int_e \phi \nabla \cdot \mathbf{F}' \, dx = \int_e \phi (D' - D) \, dx, \quad \forall \phi \in \mathbb{V}_2(e), \quad (65)$$

$$\int_e \mathbf{w} \cdot \mathbf{F}' + \nabla \cdot \mathbf{w} p \, dx = 0, \quad \forall \mathbf{w} \in \mathbb{V}_1(e), \quad (66)$$

subject to the above zero Dirichlet boundary conditions for \mathbf{F}' , where p is introduced to determine a unique \mathbf{F}' .

2.3.3. Advection scheme for velocity \mathbf{u}

The advection scheme described in this section follows the following design strategy: find an advection scheme for q that is compatible with equation (37), and select the corresponding \mathbf{Q} for insertion into that equation to compute $R_{\mathbf{u}}[\mathbf{w}]$. Selecting $\mathbf{w} = -\nabla^\perp \gamma$ in equation (37), evaluating equation (17) at time levels n and $n+1$ and substituting gives

$$\int_\Omega \gamma q^{n+1} D^{n+1} \, dx - \int_\Omega \gamma q^n D^n \, dx - \Delta t \int_\Omega \nabla \gamma \cdot \mathbf{Q} \, dx = 0, \quad \forall \gamma \in \mathbb{V}^0, \quad (67)$$

after noting that \mathbf{w} is divergence-free. McRae and Cotter [24] used $\mathbf{Q} = \bar{\mathbf{F}} q^{n+1/2}$ to obtain an implicit midpoint rule time discretisation for an energy–enstrophy conserving formulation. In this paper, we aim to use higher-order stabilised advection schemes in order to obtain accurate representation of potential vorticity transport. We note that Streamline Upwind Petrov Galerkin methods [10] and Taylor–Galerkin methods [14] all result in time-discrete formulations equivalent to equation (67).

It is desirable to use a higher order timestepping scheme for q , to obtain accurate transport of potential vorticity using the balanced flow. In particular, odd-ordered schemes are attractive since the leading order error is diffusive rather than dispersive. Hence, in this paper we make use of the two-step third-order unconditionally-stable Taylor–Galerkin scheme of Safjan and Oden [30].

Taylor–Galerkin schemes are built by transforming time derivatives into space derivatives using the advection equation.

The general form of a multistage Taylor–Galerkin method is

$$Z_i - \eta(\Delta t)^2(Z_i)_{tt} = Z^n + \Delta t \sum_{j=1}^i \mu_{ij}(Z_j)_t + (\Delta t)^2 \sum_{j=1}^i v_{ij}(Z_j)_{tt}, \quad i = 1, \dots, k, \quad (68)$$

where η is a stabilisation parameter, the subscript i is the stage index, and the $\{\mu\}_{ij}$ and $\{v\}_{ij}$ are coefficients defined in Safjan and Oden [30]. After computing these stage variables, the value of Z_k is copied into Z^{n+1} .

In our discretisation we use a formulation where $Z = qD$. Then we have

$$\int_{\Omega} \gamma(qD)_t dx = \int_{\Omega} \nabla \gamma \cdot (\bar{\mathbf{F}}q) dx, \quad \forall \gamma \in \mathbb{V}_0, \quad (69)$$

$$\int_{\Omega} \gamma(qD)_{tt} dx = \int_{\Omega} \nabla \gamma \cdot (\bar{\mathbf{F}}q_t) dx = - \int_{\Omega} \nabla \gamma \cdot \left(\bar{\mathbf{F}} \frac{\bar{\mathbf{F}}}{D} \cdot \nabla q \right) dx, \quad \forall \gamma \in \mathbb{V}_0, \quad (70)$$

recalling that $\bar{\mathbf{F}} = \bar{D}\bar{\mathbf{u}}$ is considered to be time-independent over the advection step as part of the semi-implicit discretisation. Combination with Equation (68) leads to

$$\begin{aligned} \int_{\Omega} \gamma q_i D^{n+1} + \eta \Delta t^2 \frac{\bar{\mathbf{F}}}{D} \cdot \nabla \gamma \bar{\mathbf{F}} \cdot \nabla q_i dx &= \int_{\Omega} \gamma q^n D^n + \Delta t \sum_{j=1}^i \mu_{ij} \int_{\Omega} \nabla \gamma \cdot \bar{\mathbf{F}} q_j dx - \\ &\quad (\Delta t)^2 \sum_{j=1}^i v_{ij} \int_{\Omega} \frac{\bar{\mathbf{F}}}{D} \cdot \nabla \gamma \bar{\mathbf{F}} \cdot \nabla q_j dx, \quad \forall \gamma \in \mathbb{V}_0. \end{aligned} \quad (71)$$

Note that this equation involves solving a Helmholtz-type equation with derivatives in the streamline direction for each stage variable. This equation is symmetric positive definite and well-conditioned for $\mathcal{O}(1)$ Courant numbers; the conjugate gradient method converges quickly with simple SOR preconditioning.

In this paper we took the following coefficients

$$\begin{aligned} \eta &= 0.48, \quad c_1 = \frac{1}{2} \left(1 + \left(-\frac{1}{3} + 8\eta \right)^{\frac{1}{2}} \right), \quad \mu_{11} = c_1, \quad \mu_{12} = 0, \quad \mu_{21} = \frac{1}{2} \left(3 - \frac{1}{c_1} \right), \\ \mu_{22} &= \frac{1}{2} \left(\frac{1}{c_1} - 1 \right), \quad v_{11} = \frac{1}{2} c_1^2 - \eta, \quad v_{12} = 0, \quad v_{21} = \frac{1}{4} (3c_1 - 1) - \eta, \quad v_{22} = \frac{1}{4} (1 - c_1). \end{aligned} \quad (72)$$

Safjan and Oden [30] showed this scheme to be third order and unconditionally stable for $\eta > 0.473$.

Having solved for q^{n+1} , we take $i = 2$, and notice that equation (71) takes the form of equation (67), and hence \mathbf{Q} can be extracted for insertion into equation (37). For the case of curved elements, D must be replaced by \bar{D}/τ throughout.

3. Numerical results

In this section we show numerical results from three standard test cases on icosahedral grids, using the spaces (P_3, BDM_2, DG_1) , and a piecewise cubic approximation to the surface of the sphere. The code was implemented using the Firedrake finite element framework, which permits the symbolic implementation of the mixed finite element techniques discussed in this paper. Using the Unified Form Language (see Alnæs et al. [1]), which provides a high-level mathematical description of the finite element problem, code is automatically generated which forms the resulting matrix equations by assembling the local contributions from each cell and/or facet of the mesh. These equations are provided directly to PETSc [6,5], which provides direct access to runtime configurable iterative solvers and preconditioners. The hybridisation of the implicit system for the linear updates is implemented using the Slate framework [16], which performs the local elimination and recovery operations. The reduced equation for the trace variables is numerically inverted using the conjugate gradient method and PETSc's smooth aggregation multigrid preconditioner (GAMG).

The first two test cases are numbers 2 (solid body rotation) and 5 (flow over a mountain) from Williamson et al. [36] and the third is the barotropically unstable jet from Galewsky et al. [15]. Table 1 contains information on the properties of the 4 grids that we use for the Williamson convergence tests, along with the timesteps used. The timestep was chosen to give a constant Courant number across the different resolutions; 0.2 in the solid body rotation test and 0.06 in the flow over a mountain test, although we note that in order to see second order convergence, we had to further reduce the timestep for the highest resolution mountain simulation because at this resolution the time discretisation errors start to dominate. Fig. 1 shows the lowest resolution icosahedral grid that we use.

Table 1

Grid properties for the 4 grids used in the convergence tests, including the number of degrees of freedom (DOFs) for velocity \mathbf{u} , depth D and potential vorticity q , along with the timestep used for the solid body rotation test (2) and flow over a mountain test (5).

Grid properties				DOFs			Test 2	Test 5
cells	nodes	Δx_{\max} (km)	Δx_{\min} (km)	\mathbf{u}	D	q	Δt (s)	Δt (s)
1280	642	1054	720	9600	3840	5762	3000	900
5120	2562	527	348	38400	15360	23042	1500	450
20480	10242	263	171	153600	61440	92162	750	225
81920	40962	132	85	614400	245760	368642	375	84.375

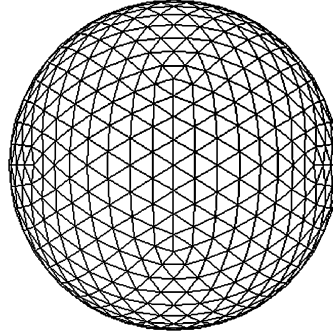


Fig. 1. Icosahedral sphere grid, viewed looking down on the North pole, corresponding to the lowest resolution described in Table 1.

Table 2

Normalised depth, D , and velocity, \mathbf{u} , errors at day 15 for the solid body rotation test case.

cells	$L_2(D)$	$L_\infty(D)$	$L_2(\mathbf{u})$	$L_\infty(\mathbf{u})$
1280	5.929×10^{-5}	2.177×10^{-4}	7.180×10^{-4}	1.701×10^{-3}
5120	9.154×10^{-6}	4.405×10^{-5}	1.261×10^{-4}	2.978×10^{-4}
20480	1.840×10^{-6}	1.062×10^{-5}	2.761×10^{-5}	6.588×10^{-5}
81920	4.288×10^{-7}	2.636×10^{-6}	6.640×10^{-6}	1.551×10^{-5}

Where shown, normalised errors in a field q are computed as in Williamson et al. [36] as

$$L_2(q) = \frac{(\int_{\Omega} (q - q_T)^2 dV)^{\frac{1}{2}}}{(\int_{\Omega} q_T^2 dV)^{\frac{1}{2}}}, \quad (73)$$

$$L_\infty(q) = \frac{\max |q - q_T|}{\max |q_T|}, \quad (74)$$

where q_T is the true solution, specified either analytically (as in the solid body rotation test) or from a high resolution reference solution (as in the flow over a mountain test).

3.1. Solid body rotation (Williamson, test case 2)

This test case is initialised with depth and velocity fields that are in geostrophic balance:

$$D = D_0 - \left(R\Omega u_0 + \frac{u_0^2}{2} \right) \frac{z^2}{gR^2}, \quad (75)$$

$$\mathbf{u} = \frac{u_0}{R}(-y, x, 0), \quad (76)$$

where $R = 6.37122 \times 10^6$ m is the radius of the Earth, $\Omega = 7.292 \times 10^{-5} \text{ s}^{-1}$ is the rotation rate of the Earth, (x, y, z) are the 3D Cartesian coordinates, $g = 9.80616 \text{ m s}^{-2}$ is the gravitational acceleration, $D_0 = 2.94 \times 10^4 / g \approx 2998$ m and $u_0 = 2\pi R / (12 \text{ days}) \approx 38.6 \text{ m s}^{-1}$. Since there is no topography or other forcing, the flow should remain in this steady, balanced state. As the flow is steady we have an analytic solution which allows us to compute errors and hence an order of convergence for our method. We present these results in Table 2 and Fig. 2 reveals that our method is converging at second order. The depth error at day 15, shown in Fig. 3, is large scale and shows some evidence of grid imprinting.

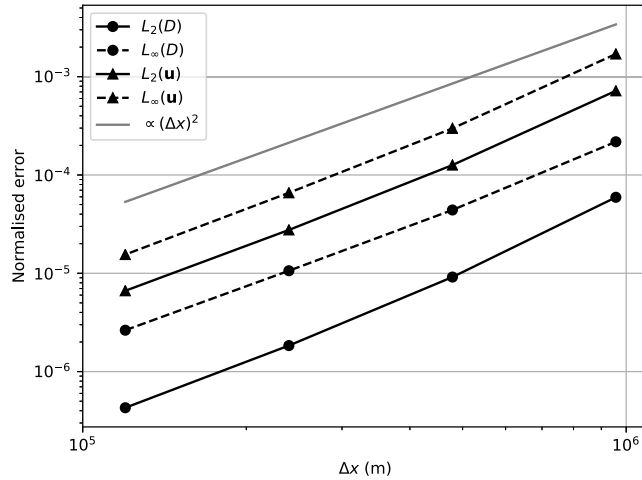


Fig. 2. Solid body rotation test case: normalised depth, D , and velocity, u , errors at day 15 versus average mesh size Δx .

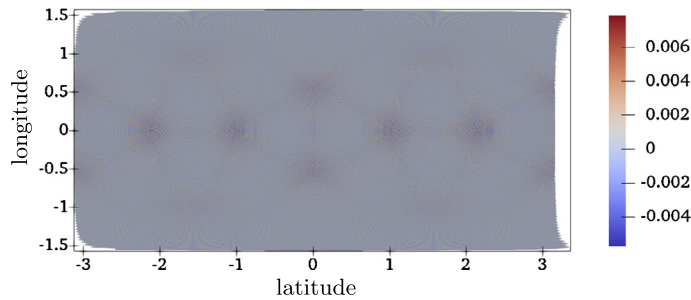


Fig. 3. Solid body rotation test case: depth error (in metres) at day 15. The range is min: -5.8×10^{-3} m, max: 7.9×10^{-3} m.

Table 3

Normalised depth errors at day 15 for the flow over a mountain test case.

cells	$L_2(D)$	$L_\infty(D)$
1280	1.406×10^{-3}	7.963×10^{-3}
5120	6.776×10^{-4}	3.317×10^{-3}
20480	2.136×10^{-4}	1.200×10^{-3}
81920	4.159×10^{-5}	2.989×10^{-4}

3.2. Flow over a mountain (Williamson, test case 5)

This test case uses the same zonal flow initial conditions (75)–(76) as the previous test but with $D_0 = 5960$ m and $u_0 = 20$ m. An isolated, conical mountain, given by

$$D_s = D_{s0}(1 - r/R_0) \quad (77)$$

with $D_{s0} = 2000$ m, $R_0 = \pi/9$ and $r^2 = \min[R_0^2, (\phi - \phi_c)^2 + (\lambda - \lambda_c)^2]$ is placed with its centre at latitude $\phi = \pi/6$ and longitude $\lambda = -\pi/2$. As the zonal flow interacts with the mountain, it produces waves that travel around the globe. As there is no analytical solution for this problem, the model output at 15 days is compared to a high resolution (a 2048 by 1024 grid, with a timestep of 45 s) reference solution generated using a semi-Lagrangian shallow water code provided by John Thuburn. We present the magnitude of the errors in Table 3 and Fig. 4 reveals that our method is converging at second order. We plot the depth errors at day 15 in Fig. 5. We can see that the error is small scale and is not dominated by errors due to grid imprinting.

Up until this point the flow is only weakly nonlinear (see Fig. 6) so, as in Thuburn et al. [33], we continued the highest resolution simulation until day 50. By this time, fine scale structure has been generated as the flow becomes more nonlinear; the PV field develops sharp gradients and filaments that stretch out and roll up. These features can be seen in the potential vorticity field at 50 days, shown in Fig. 7.

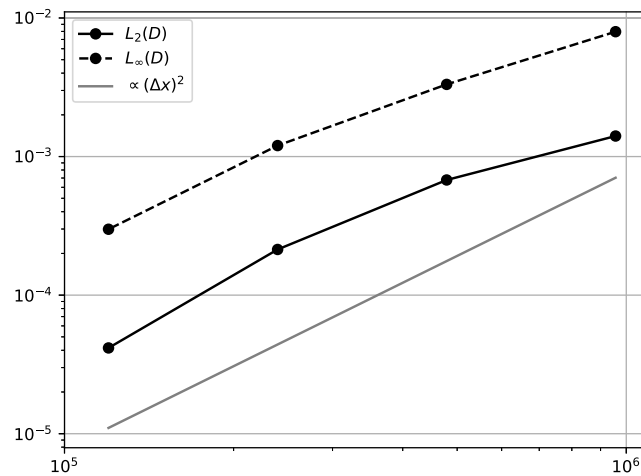


Fig. 4. Flow over a mountain test case: normalised depth errors at day 15 versus average mesh size Δx .

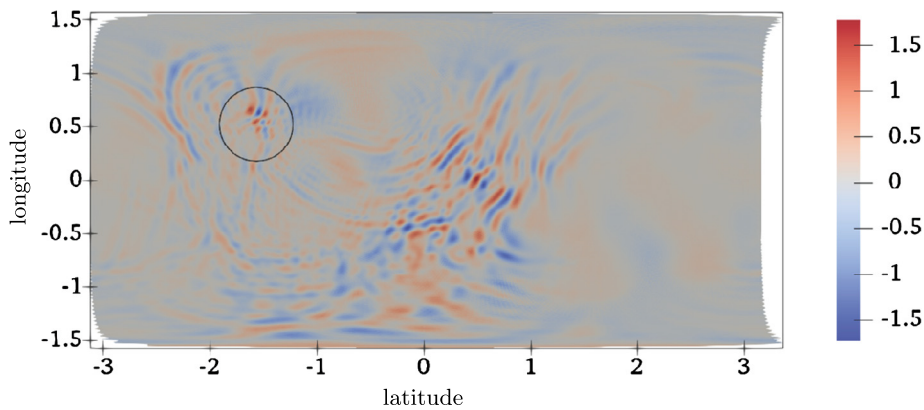


Fig. 5. Flow over a mountain test case: depth errors (in metres) at day 15. The range is min: -1.74 m, max: 1.78 m.

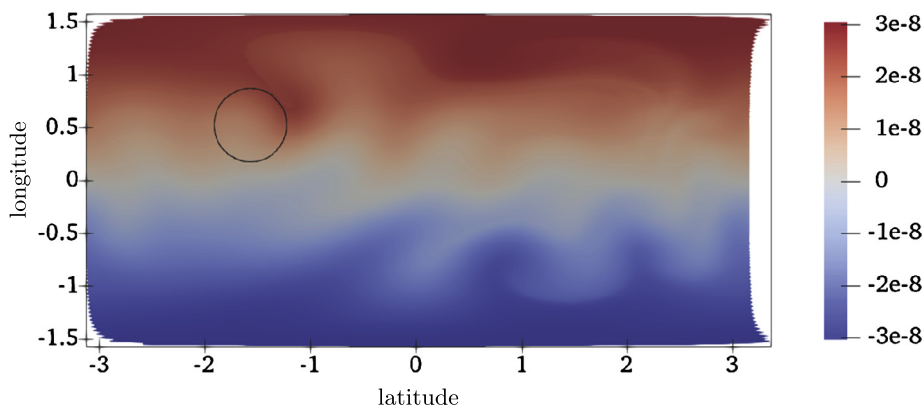


Fig. 6. Flow over a mountain test case: potential vorticity at day 15. The range is min: $-3.05 \times 10^{-8} \text{ (ms)}^{-1}$, max: $3.05 \times 10^{-8} \text{ (ms)}^{-1}$. The circle indicates the position of the mountain.

3.3. Barotropically unstable jet (Galewsky)

The details of this test case are specified in Galewsky et al. [15]. The initial condition consists of a strong midlatitude jet, with an added perturbation, which is barotropically unstable and evolves to produce vortices and small scale structure. It has become a particularly useful test for models on grids that are not aligned with latitude/longitude because these grids can induce early development of the instability and even lead to the final solution (at day 6) having the incorrect wavenumber.

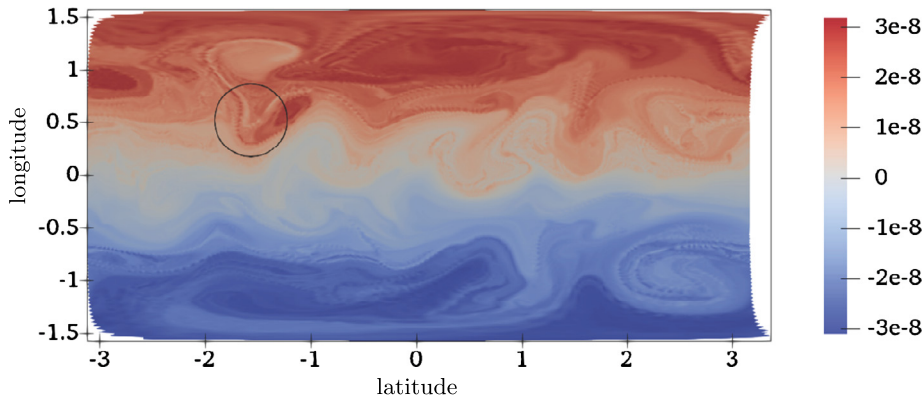


Fig. 7. Flow over a mountain test case: potential vorticity at day 50. The range is min: $-3.1 \times 10^{-8} \text{ (ms)}^{-1}$, max: $3.2 \times 10^{-8} \text{ (ms)}^{-1}$. The circle indicates the position of the mountain.

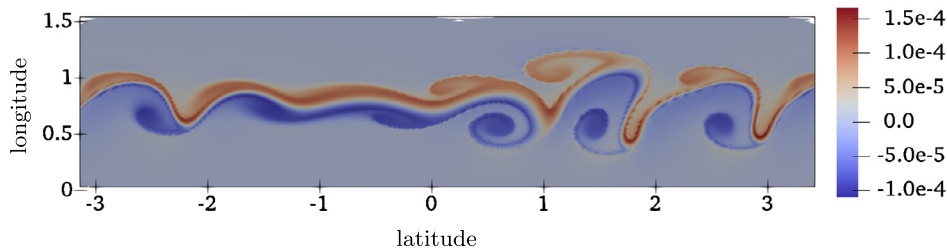


Fig. 8. Snapshot of the northern hemisphere relative vorticity field at day 6 for the Galewsky jet test. The range is min: $-1.10 \times 10^{-4} \text{ (ms)}^{-1}$, max: $1.66 \times 10^{-4} \text{ (ms)}^{-1}$.

Table 4

Values of Q (see equation (78)) for the Galewsky test case.

Time (days)	Q	Q
0	-5.70×10^{-15}	Min: -1.80×10^{-14}
2	-5.22×10^{-15}	Max: 1.73×10^{-14}
4	1.05×10^{-14}	Mean: 4.23×10^{-16}
6	-2.02×10^{-16}	Standard deviation: 4.36×10^{-15}

Again, there is no analytic solution so results are compared to those in the literature (see for example Thuburn et al. [33] and Weller [35]). An important feature to reproduce is the relatively straight path of the jet across approximately quarter of the globe – this is not seen in models where grid imprinting has resulted in the generation of instability along the length of the jet [35].

Fig. 8 shows the vorticity field, computed on the highest resolution grid (see Table 1) with timestep of 120 s, in the Northern hemisphere after 6 days. The maximum Courant number reached during this simulation is 0.28. We note that the jet has the correct wavenumber and the expected quiescent section. Lower resolution simulations, as others have found [33], did not have these requisite features. The conservation properties of the algorithm are demonstrated in Table 4 which gives values of the normalised integral of the potential vorticity over the sphere,

$$Q = \frac{\int_{\Omega} q D \, dV}{\|q_0\|_{L^2} \|D_0\|_{L^2}} \quad (78)$$

where q_0 and D_0 are the initial values of the potential vorticity and depth respectively, and $\|\cdot\|_{L^2}$ indicates the (un-normalised) L^2 norm. This is conserved (and zero) by construction (see equation 19 and the following discussion).

4. Summary and outlook

We have built upon the work of Cotter and Shipton [11] and McRae and Cotter [24] to produce a semi-implicit compatible finite element model for the nonlinear rotating shallow water equations on the sphere. The important features are that we introduce higher-order upwind advection schemes that maintain PV conservation, and consistency of PV advection with

the mass conservation law. By applying the model to standard test cases we have demonstrated that this model has the expected second order convergence rate and can produce the required features of both large scale balanced flows and unstable turbulent flows. The developments of this paper inform our ongoing development of a three dimensional dynamical core on the sphere.

Acknowledgements

The authors would like to thank John Thuburn for providing the semi-Lagrangian code that was used for the Williamson 5 convergence test, and the Firedrake project [27,23,21,22,16], along with PETSc [6,5] and other upstream tools [19,13] for making the code development in this paper possible. Jemma Shipton and Colin Cotter were supported by Natural Environment Research Council grants NE/M013634/1 and NE/K006789/1 and Engineering and Physical Sciences Research Council grant EP/L000407/1 and Thomas Gibson would like to acknowledge funding from the Engineering and Physical Sciences Research Council Centre for Doctoral Training in Mathematics of Planet Earth, EP/L016613/1.

References

- [1] Martin S. Alnæs, Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, Garth N. Wells, Unified form language: a domain-specific language for weak formulations of partial differential equations, *ACM Trans. Math. Software* 40 (2) (2014) 9.
- [2] Akio Arakawa, Vivian R. Lamb, A potential enstrophy and energy conserving scheme for the shallow water equations, *Mon. Weather Rev.* 109 (1) (1981) 18–36.
- [3] D.N. Arnold, R.S. Falk, R. Winther, Finite element exterior calculus, homological techniques, and applications, *Acta Numer.* 15 (1) (2006) 1–155.
- [4] D.N. Arnold, D. Boffi, F. Bonizzoni, Finite element differential forms on curvilinear cubic meshes and their approximation properties, *Numer. Math.* (2014), <https://doi.org/10.1007/s00211-014-0631-3>, arXiv:1204.2595.
- [5] Satish Balay, William D. Gropp, Lois Curfman McInnes, Barry F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in: E. Arge, A.M. Bruaset, H.P. Langtangen (Eds.), *Modern Software Tools in Scientific Computing*, Birkhäuser Press, 1997, pp. 163–202.
- [6] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Karl Rupp, Barry F. Smith, Stefano Zampini, Hong Zhang, Hong Zhang, PETSc Users Manual, Technical Report ANL-95/11 – Revision 3.7, Argonne National Laboratory, 2016.
- [7] P.B. Bochev, D. Ridzal, Rehabilitation of the lowest-order Raviart–Thomas element on quadrilateral grids, *SIAM J. Numer. Anal.* 47 (1) (2008) 487–507.
- [8] D. Boffi, L. Gastaldi, Some remarks on quadrilateral mixed finite elements, *Comput. Struct.* 87 (11) (2009) 751–757.
- [9] D. Boffi, F. Brezzi, M. Fortin, *Mixed Finite Element Methods and Applications*, Springer, 2013.
- [10] Alexander N. Brooks, Thomas J.R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations, *Comput. Methods Appl. Mech. Engrg.* 32 (1) (1982) 199–259.
- [11] C.J. Cotter, J. Shipton, Mixed finite elements for numerical weather prediction, *J. Comput. Phys.* 231 (21) (2012) 7076–7091.
- [12] C.J. Cotter, J. Thuburn, A finite element exterior calculus framework for the rotating shallow-water equations, *J. Comput. Phys.* 257 (2014) 1506–1526.
- [13] Lisandro D. Dalcin, Rodrigo R. Paz, Pablo A. Kler, Alejandro Cosimo, Parallel distributed computing using Python, *Adv. Water Resour.* 34 (9) (2011) 1124–1139, <https://doi.org/10.1016/j.advwatres.2011.04.013>.
- [14] Jean Donea, A Taylor–Galerkin method for convective transport problems, *Internat. J. Numer. Methods Engrg.* 20 (1) (1984) 101–119.
- [15] J. Galewsky, R.K. Scott, L.M. Polvani, An initial-value problem for testing numerical models of the global shallow-water equations, *Tellus A* 56 (5) (2004) 429–440.
- [16] Thomas H. Gibson, Lawrence Mitchell, David A. Ham, Colin J. Cotter, A domain-specific language for the hybridization and static condensation of finite element methods, arXiv:1802.00303, 2018.
- [17] Jayadeep Gopalakrishnan, Shuguang Tan, A convergent multigrid cycle for the hybridized mixed method, *Numer. Linear Algebra Appl.* 16 (9) (2009) 689–714.
- [18] Sigal Gottlieb, Chi-Wang Shu, Eitan Tadmor, Strong stability-preserving high-order time discretization methods, *SIAM Rev.* 43 (1) (2001) 89–112.
- [19] Bruce Hendrickson, Robert Leland, A multilevel algorithm for partitioning graphs, in: *Supercomputing '95: Proceedings of the 1995 ACM/IEEE Conference on Supercomputing*, ACM Press, New York, 1995, p. 28.
- [20] M. Holst, A. Stern, Geometric variational crimes: Hilbert complexes, finite element exterior calculus, and problems on hypersurfaces, *Found. Comput. Math.* 12 (3) (2012) 263–293.
- [21] Miklós Homolya, Lawrence Mitchell, Fabio Luporini, David A. Ham, TSFC: a structure-preserving form compiler, arXiv:1705.03667, 2017.
- [22] Robert C. Kirby, Lawrence Mitchell, Solver composition across the PDE/linear algebra barrier, *SIAM J. Sci. Comput.* 40 (1) (2018) C76–C98, <https://doi.org/10.1137/17M1133208>.
- [23] Fabio Luporini, David A. Ham, Paul H.J. Kelly, An algorithm for the optimization of finite element integration loops, *ACM Trans. Math. Software* 44 (1) (2017) 3:1–3:26, <https://doi.org/10.1145/3054944>.
- [24] Andrew T.T. McRae, Colin J. Cotter, Energy- and enstrophy-conserving schemes for the shallow-water equations, based on mimetic finite elements, *Q. J. R. Meteorol. Soc.* 140 (684) (2014) 2223–2234.
- [25] A. Natale, J. Shipton, C.J. Cotter, Compatible finite element spaces for geophysical fluid dynamics, *Dyn. Stat. Climate Syst.* 1 (1) (2016), <https://doi.org/10.1093/climsys/dzw005>.
- [26] W.M. Putman, S.J. Lin, Finite-volume transport on various cubed-sphere grids, *J. Comput. Phys.* 227 (1) (2007) 55–78.
- [27] Florian Rathgeber, David A. Ham, Lawrence Mitchell, Michael Lange, Fabio Luporini, Andrew T.T. McRae, Gheorghe-Teodor Bercea, Graham R. Markall, Paul H.J. Kelly, Firedrake: automating the finite element method by composing abstractions, *ACM Trans. Math. Software* 43 (3) (2016) 1–27, <https://doi.org/10.1145/2998441>, arXiv:1501.01809.
- [28] T.D. Ringler, J. Thuburn, J.B. Klemp, W.C. Skamarock, A unified approach to energy conservation and potential vorticity dynamics for arbitrarily-structured C-grids, *J. Comput. Phys.* 229 (9) (2010) 3065–3090, <https://doi.org/10.1016/j.jcp.2009.12.007>.
- [29] M.E. Rognes, D.A. Ham, C.J. Cotter, A.T.T. McRae, Automating the solution of PDEs on the sphere and other manifolds in FEniCS 1.2, *Geosci. Model Devel. Discuss.* 6 (3) (2013) 3557–3614.
- [30] A. Safjan, J.T. Oden, High-order Taylor–Galerkin and adaptive h–p methods for second-order hyperbolic systems: application to elastodynamics, *Comput. Methods Appl. Mech. Engrg.* 103 (1) (1993) 187–230.
- [31] Andrew Staniforth, John Thuburn, Horizontal grids for global weather and climate prediction models: a review, *Q. J. R. Meteorol. Soc.* 138 (662) (2012) 1–26.

- [32] J. Thuburn, C.J. Cotter, A framework for mimetic discretization of the rotating shallow-water equations on arbitrary polygonal grids, *SIAM J. Sci. Comput.* 34 (3) (2012) B203–B225.
- [33] J. Thuburn, C.J. Cotter, T. Dubos, A mimetic, semi-implicit, forward-in-time, finite volume shallow water model: comparison of hexagonal–icosahedral and cubed sphere grids, *Geosci. Model Devel. Discuss.* 6 (2013) 6867–6925.
- [34] John Thuburn, Colin J. Cotter, A primal–dual mimetic finite element scheme for the rotating shallow water equations on polygonal spherical meshes, *J. Comput. Phys.* 290 (2015) 274–297.
- [35] Hilary Weller, Non-orthogonal version of the arbitrary polygonal c-grid and a new diamond grid, *Geosci. Model Devel.* 7 (2013) 779–797.
- [36] D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, P.N. Swarztrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, *J. Comput. Phys.* 102 (1) (1992) 211–224.